Network Coding Based Cooperative Repair for Cache-enabled Wireless Network

Cheng Zhan[1], Zhe Wen

School of Computer and Information Science, Southwest University, P.R. China
zhanc@swu.edu.cn, wenzheswu@gmail.com

Abstract

In wireless coded cache network, data contents are cached in a number of mobile devices using an erasure correcting code, and thus users can retrieve data contents from other mobile devices using device-to-device communication. In this paper, we consider the repair problem when multiple devices that cache data contents fail or leave the network. By exploiting the wireless broadcast nature, we formulate the repair problem over the broadcast channels using an integer linear programming formulation, aiming at minimizing the number of necessary broadcast transmissions. We also study the construction of repair codes and propose a decentralized repair coding method. Simulation results show that the performance using our method outperforms that of traditional cooperative repair scheme, which is the basic repair method for wired distributed storage systems.

Keywords: Wireless distributed caching, Network coding, Repair scheme

1 Introduction

With the increasing popularity of smart mobile devices, mobile traffic will boom in the near future. Mobile data hungry applications, such as audio and video streaming, cloud-based services and social sharing, are more and more popular [1-2]. Adding traffic beyond a certain limit will reduce the quality of service (QoS) perceived by the users. Therefore, this dramatic increase in demand poses a challenge on the wireless mobile networks.

One of the recent suggestions to enhance the spectrum efficiency in the 5G architecture is to bring the content especially the popular files closer to the clients, by deploying a large number of low-cost wireless nodes with large storage capacity. These nodes are usually referred to as storage nodes [3-4]. It was suggested in [5] to store content directly in the mobile devices, taking advantage of the high storage capacity of modern smart phones. The requested content can then be directly retrieved from neighboring mobile devices, using device-to-device (D2D) communication. This allows for a more efficient content delivery without additional infrastructure cost. Caching in the mobile devices to alleviate the wireless bottleneck has attracted a significant interest in the research community in the recent years [6].

A relevant problem in D2D-assisted mobile caching networks is the repairing of the lost data contents when a caching device is unavailable, e.g., when a caching device fails or leaves the network. In distributed storage systems, reparability is an important design issue. The work [7] introduced MDS code and has stimulated a lot of study on efficient repair scheme of failed nodes in wired distributed storage systems [8]. The data contents can be encoded by an (n, k) maximum distance separable (MDS) code ($n \geq k$). Specifically, the content is split into k fragments and encoded into n fragments such that any k fragments can be used to reconstruct the original file (MDS property). Most of these works focused on single-node repair scheme, meaning that nodes are assumed to be failed one by one and the repair process is triggered immediately when a node failure occurs. Moreover, there is also some work focusing on cooperative repair scheme [9] when considering multiple failed nodes.

Although designs for traditional distributed storage can also be applied to wireless network, it is important to understand the fundamental difference when considering the repair problem in wireless caching networks. The basic characteristic of the wireless medium is its broadcast nature. The work in [10] derived analytical expressions for the overall communication cost of repair and download in wireless D2D network as a function of the repair interval using the traditional wired MDS code. The work in [11] studied the repair problem in wireless storage network when parts of stored packets in nodes are lost. However, in practice, if a caching device fails or leaves the network, all data cached in the device need to be repaired.

Considering a wireless coded caching system as shown in Figure 1. The caching network contains four caching devices. These devices are supposed to store a file containing four packets $\{a_1, a_2, b_1, b_2\}$ encoded by

*Corresponding Author: Cheng Zhan; E-mail: zhanc@swu.edu.cn
DOI: 10.3966/160792642020072104019

a (4, 2)-MDS code. Suppose that node 2 and node 4 leave the network simultaneously, our aim is to replace them by new devices, called the newcomers, i.e., node 5 and node 6. Using the cooperative regenerating codes mentioned in [9], node 5 gets a_1, a_1+b_1 and b_2, while node 6 gets a_2, $2a_2$+b_2 and $2a_1$+b_1. Node 5 can obtain b_1 and b_2 by decoding theses inputs by MDS decoding operations. Likewise, node 6 can obtain a_2+b_2 and $2a_1$+b_1. The data contents of the newcomers are regenerated after six transmissions (See Figure 1(a)). However, if we exploit the broadcast nature of the wireless medium during the repair process, the number of transmissions will be reduced. As shown in Figure 1(b), node 1 broadcasts a_1 and a_2, while node 3 broadcasts a_1+b_1 and $2a_2$+b_2. Node 5 can decode out b_1 and b_2 by performing linear operations (a_1+b_1)-a_1 and $2a_2$+b_2-$2a_2$ separately. Similarly, node 6 can obtain $2a_1$+b_1 and a_2+b_2. Therefore only four broadcast transmissions are needed.

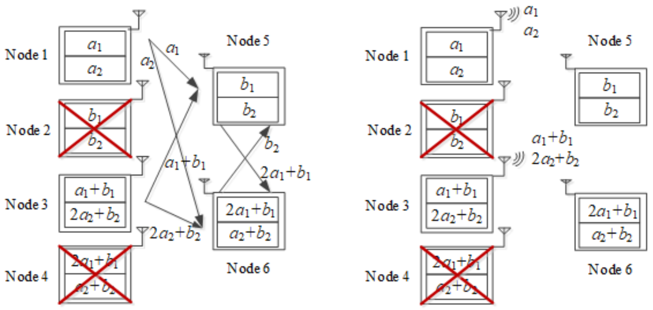

Figure 1. Wireless coded storage using (4, 2)-MDS code

Inspired by the above example, we consider the efficient repair scheme for the wireless coded storage networks, broadcast nature of the communication channel is investigated during the repair process when there are more than one failed devices. Similar to previous wired repair problems [7], the repaired packets might not be the same as the lost packets but the MDS property of the system is maintained. Our aim is to minimize the number of transmissions for devices to repair. The main contributions of this paper are summarized as follows:

- We study the repair scheme when multiple devices that stores data fail or leave the network by exploiting the wireless broadcast nature using an information flow graph model.
- We formulate the repair problem over the broadcast channels using an integer linear programming formulation, aiming at reducing the number of necessary transmissions.
- We study the construction of repair codes and propose a decentralized repair coding method. Simulations results demonstrate significant performance benefits in terms of number of transmissions.

The remainder of this paper is organized as follows. In Section 2, we will give the system model and problem statement. In Section 3, we will formulate the problem as an integer linear programming using an information flow graph model. The heuristic data repair algorithm for maintaining the MDS property with random linear coding will be presented in Section 4. Simulation results are shown in Section 5. Finally, we conclude the paper in Section 6.

2 System Model and Problem Description

We consider the scenario which consists of a base station (BS), and n mobile caching devices, $D=\{d_1, d_2, ..., d_n\}$. We assume that each caching device can store c packets at most, and devices are fully connected by a wireless broadcast medium. The scenario is usually applied when a group of smartphone users, within proximity of each other, share the same data contents with applications ranging from multimedia content sharing, group multicast, and public safety [12].

At the initial stage, the base station divide a file to k fragments and each fragment is then further equally divided to c packets, therefore there are ck packets in total. We apply an (n, k)-MDS code on these packets and then generate cn encoded packets, which are stored on n caching nodes such that the any user (or data collector) can retrieve the original file from any k devices. These n caching devices are not reliable and can fail or leave the network. The set of failed nodes is denoted as F. When the number of failed nodes is accumulated up to a threshold γ, $\gamma \geq 1$, the repair process is triggered. During the repair process, γ new devices which are selected as caching devices, will join the system. We use N to denote the set of newcomers. Therefore n-γ active caching devices will broadcast packets to the newcomers. We assume that $n-\gamma \geq k$ so that the repair is always feasible. In other words, the total number of available encoded packets, i.e., $c(n\text{-}\gamma)$, must be greater than or equal to that of the original file, i.e, ck.

It is worthwhile to note that the information about status (active or fail) of a device can be collected by BS with active detection. Furthermore, it is difficult to obtain the failure model and network left model of the devices. For simplicity, we assume that every device fail or leave the network independently and identically. The more accurate model for failure model and network left model in repair design is non-trivial, which will be left as our future work.

We consider the interference model in [13]: each device can either transmit or receive through WiFi, and all transmissions in the range of the receiver are considered interfering. Since we consider a group of devices within proximity of each other, each device belongs to the transmitting coverage zones of other devices, the D2D network is a fully connected D2D network. Thus any transmission in the local area interferes with any other transmission, which means that only one device can transmit at each time slot.

In order to maintain the MDS property, n-γ active caching devices will broadcast encoded packets to the newcomers. In this setting, we aim to minimize the total number of transmissions. It is crucial to determine which caching device should transmit and what encoded packets should be transmitted. In this paper, the problem is that given the set of caching devices D, the set of failed devices F, and the set of newcomers N, how to encode and transmit packets in each transmission slot to satisfy the MDS property. The objective is to minimize the number of transmissions. Such an encoding and transmission decision problem is referred to as *Wireless D2D Repair Code (WDRC) problem*. Table 1 shows the notations to be used in constructing the graph model and the proposed encoding algorithm.

Table 1. Notation

Symbol	Description
n	Number of caching devices
D	The set of caching devices
d_i	User device i
c	Number of packets stored at each caching device
k	Number of fragments divided by a file
F	The set of failed nodes
γ	The threshold of the number of failed nodes
N	The set of newcomers
R_S	The set of devices that a data collector connects to
t_i	The number of transmissions from device d_i

3 Wireless D2D Repair Code Problem

In the following, we will first develop constraint conditions of the WDRC problem using an information flow graph model, and then we will formulate WDRC problem using an integer linear programming.

3.1 Information Flow Graph Model

After the repair process, we aim to maintain the MDS property. In other words, data collector can retrieve the whole file from any k devices. We use R_S to denote the set of devices that a data collector connects to, thus, $|R_S|= k$ and $R_S \subseteq (D-F+N)$. Let t_i be the number of transmissions from device d_i via broadcast link, $1 \le i \le n$. From the following theorem, we can obtain the constraint conditions of the WDRC problem,

Theorem 1: In WDRC problem, for any R_S, $|R_S|=k$, and $R_S \subseteq (D-F+N)$, the following inequality holds,

$$c\,|R_S \cap D| + \sum\nolimits_{d_i \in D-F-R_S \cap D} t_i \ge ck.$$

Proof: We can model the WDRC problem as an information flow problem in a multicast network, which is described in a directed acyclic graph (shown in Figure 2).

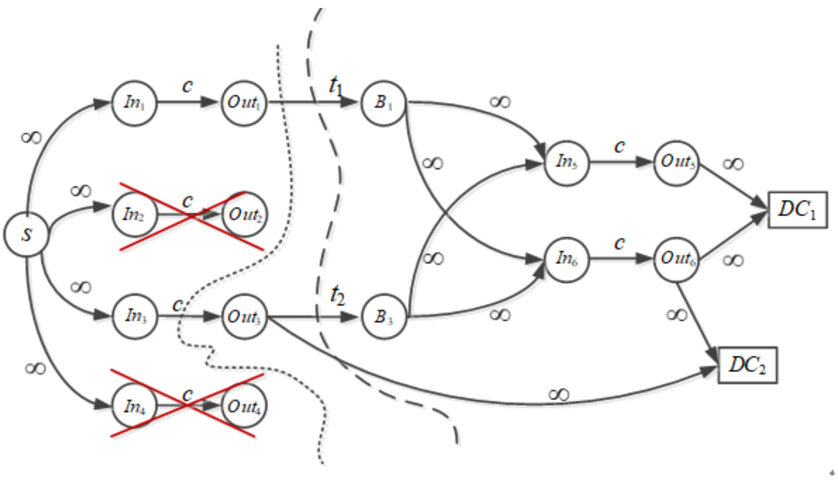

Figure 2. Information flow graph

The graph includes one virtual source node S which has access to the original file, multiple caching devices (including failed ones and newcomers), and two data collectors, denoted as DC_1 and DC_2. Each caching device d_j is represented by two vertices, i.e., in-vertex In_j, out-vertex Out_j, and a directed edge $In_j \to Out_j$ with parameter c, where c is the caching size of devices d_j. In the initialization stage that the data contents are first stored at the caching nodes, which is equivalent to the case that the source vertex S transmits packets to the caching nodes and then becomes inactive. This is modeled by adding the edges $S \to In_j$ with capacity ∞, $1 \le j \le n$.

During the repair process, device d_i broadcasts t_i packets to newcomer device d_j, which is again modeled by two vertices In_j, Out_j, and a directed edge $In_j \to Out_j$ with parameter c. Note that the index of newcomer is different from the index of the failed node being replaced. Each of these $n - \gamma$ original devices, which we call helper device, will broadcast encoded packets for repairing. For helper device d_i, we add an auxiliary vertex, which is denoted as B_i, to which Out_i is connected by an edge with capacity t_i. Edges with capacity ∞ are added from vertex B_i to In_j of every newcomer j. In other words, the vertex B_i is used to model the broadcast feature of the wireless channel from device d_i. Then, assume that there are $\binom{n}{k}$ data collectors, each of which is connected to k distinct caching devices by infinite capacity links. Consider a data collector which is connected to R_S caching nodes, $|R_S|=k$, and $R_S \subseteq (D - F + N)$. The cut from the source to the data collector passes the edges with a sum capacity $C = c\,|R_S \cap D| + \sum_{d_i \in D-F-R_S \cap D} t_i$, The necessary condition for the existence of a valid code (such that all data collectors can reconstruct the original file) is that the sum capacity of cut of the information flow graph, i.e., C must be greater than ck, i.e., the size of the original file. Thus, $c\,|R_S \cap D| + \sum_{d_i \in D-F-R_S \cap D} t_i \geq ck$, which concludes the proof.

Consider the example in Figure 2, the cut from the source to data collector DC_1 is $\{Out_1 \rightarrow B_1, Out_3 \rightarrow B_3\}$, thus $t_1 + t_2 \geq 2c$. The cut from the source to data collector DC_2 is $\{Out_1 \rightarrow B_1, In_3 \rightarrow Out_3\}$, therefore $t_1 + c \geq 2c$.

3.2 Integer Linear Programming Formation

Since only one device can transmit at a time slot via the broadcast link, the total number of transmissions is equal to $\sum_{d_i \in D-F} t_i$. Our objective is to minimize the number of transmissions, thus the objective of the WDRC problem is to minimize $\sum_{d_i \in D-F} t_i$. From Theorem 1, an integer linear programming (ILP) formulation of the WDRC problem is given as follows. In the ILP formation, the family of inequalities represents for the constraint conditions for repair process from Theorem 1. By solving the ILP, we can obtain a solution of WDRC problem which illustrates how much encoded packets every caching device should broadcast.

Minimize $\sum_{d_i \in D-F} t_i$

Subject to $c\,|R_S \cap D| + \sum_{d_i \in D-F-R_S \cap D} t_i \geq ck,$

$$\forall R_S \subseteq (D - F + N), |R_S| = k, \quad \textbf{(1)}$$

$$t_i \geq 0, t_i \in \mathrm{Z} \qquad 1 \leq i \leq n \quad \textbf{(2)}$$

For any feasible solution of ILP, a corresponding linear network coding strategy can be computed in polynomial time using the algorithm described in [14]. Therefore, the difficulty in solving the repair problem lies solely in finding the optimal solution to the ILP. However, ILP is NP-complete. Therefore, it is impossible to find the optimal solution of ILP within polynomial time. Although there are effective methods and software to deal with such problems for small scale scenarios. For large scale scenarios, heuristic methods is needed to find the sub-optimal solution within polynomial time.. Thus, it is practical to find a heuristic solution for WDRC problem.

4 Heuristic Solution for WDRC Problem

In this section, we will present heuristic data repair algorithm for maintaining the MDS property with random linear coding [15]. Random linear coding is a simple and powerful encoding scheme, whose performance is close to optimal throughput using a decentralized algorithm in broadcast transmission schemes. Specifically, each node in the network transmits random linear combinations of the packets that they receive, with coefficients are randomly chosen from a Galois field. It has been proved that if the field size is sufficiently large, the probability that the receiver(s) will obtain linearly independent combinations (and therefore obtain innovative information) approaches 1. It should however be noted that, although random linear network coding has excellent throughput performance, if a receiver obtains an insufficient number of packets, it is extremely unlikely that they can recover any of the original packets. This can be addressed by sending additional random linear combinations until the receiver obtains the appropriate number of packets.

Our repairing algorithm contains operations both at caching devices and the newcomer devices. Firstly, let us consider the operations at caching devices. For each device $d_i \in D - F$, it can broadcast encoded packets which can be selected from its cache randomly. When newcomer devices receive theses packets, how to store them to maintain MDS property remains a problem.

Secondly, let us consider the operations at newcomer devices. In our repairing algorithm at newcomer devices, each received encoded packet is also multiplied by a coefficient independently, and multiple encoded packets are added together (under the arithmetic of the Galois Field). We add a check operation to ensure that the coefficient selection will not cause transmission waste. For the first c-1 encoded packets, where c is the caching size of the caching device, if the related coefficients for received data are all zero, then the coefficient for the last encoded data is selected from $GF(q)$-{0}. Note that for random linear coding, if a data collector has access to any n linearly independent coded packets, it can use Gaussian elimination to decode the encoded packets and retrieve n original packets with high probability. In other words, it is possible that the data collector might not decode as long as there exist one encoded packet that is linear dependent with other encoded packets. Therefore, we need to check whether this case happens and add more transmissions to make sure the stored packets at caching devices are linear independent. When a caching device have constructed a stored encoded packet, it checks whether the packet is linear independent with the encoded packets it already stored, in which case it is called an innovative packet. If not, the caching devices will send feedback information to device $d_i \in D - F$ to ask for new packets. The procedure continues until each caching device d_i stores c innovative packets.

In summary, the details of the repairing algorithm is summarized in Figure 3, e_l, $1 \leq l \leq c$ represents the l-th encoded packet at the caching device.

To better understand the process of proposed repairing algorithm, we also give an example shown in Figure 4, nodes d_1 and d_3 broadcast data in sequence to newcomers. First, d_5 receives a_1 and upon encoding, $e_1 = a_1$, $e_2 = 2a_1$. Next, d_5 receives a_2 and $e_1 = a_1+2a_2$ and $e_2=2a_1+a_2$. After receiving a_1+b_1, $e_1= 2a_1+2a_2+b_1$ and $e_2= 4a_1+a_2+2b_1$. Finally, $e_1=2a_1+ 4a_2+b_1+b_2$ and $e_2=4a_1+7a_2+2b_1+6b_2$. In this example, for which we set $q = 8$ for $GF(q)$, it is observed that if the data collector requests any 2 caching devices, then the whole file can be reconstructed, which satisfy the MDS property.

Caching device:
1. **while** ($\exists$ innovative packet request)
2.   **for** ($d_i \in D - F - R_S \bigcap D$)
3.   Randomly select encoded packet p' from d_i;
4.   broadcast encoded packet p'
5.   **end for**
6. **end while**

Newcomer device:
1. $e_l \to 0$, $1 \leq l \leq c$

Upon reception of new encoded packet e
1. $tag \to 0$;
2. **for** ($l \to 1$ to $c-1$)
3.   Randomly generate coefficient α from $GF(q)$;
4.   **if** $\alpha \neq 0$
5. $tag \to 1$;
6. $e_l \to e_l + \alpha e$;
7. **end for**
8. **if** $tag = 1$
9.   Randomly generate coefficient α from $GF(q)$;
10. **else**
11.   Randomly generate coefficient α from $GF(q) - \{0\}$;
12. $e_c \to e_l + \alpha e$;
13. **if** (stored packets are linear dependent)
14.   send innovative packet request

Figure 3. Repairing algorithm

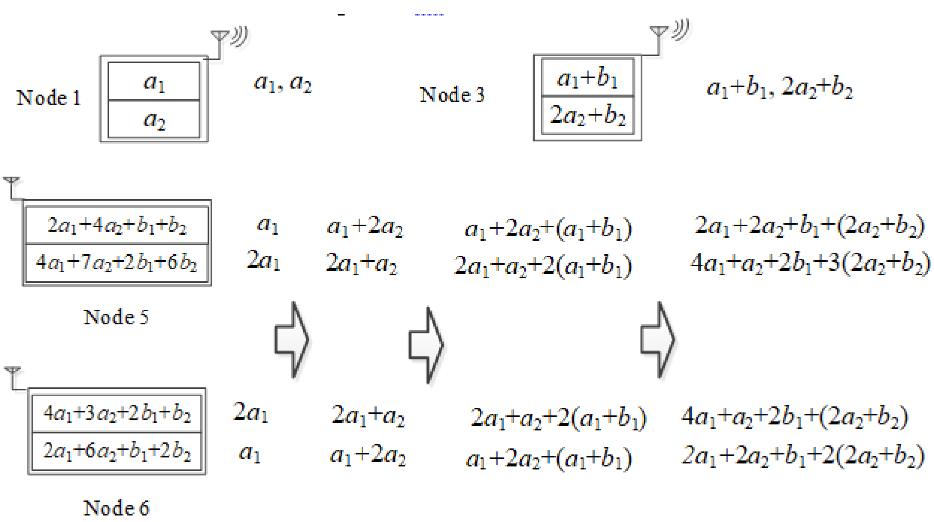

Figure 4. Example of encoding at a storage node

5 Simulations

In this section, we demonstrate the advantage of our repair scheme via simulations. We assume that there are N total devices on a small cell with size $\sqrt{N} \times \sqrt{N}$. A random subset of n devices are selected as caching devices. (n, k)-MDS code is used to cache the data content with finite field $GF(q)$, where we set $q = 2^8$. We assume that n and k are $O(\sqrt{N})$. Every caching device can store c packets, $1 \leq c \leq 10$, and devices are fully connected by a wireless broadcast medium. Failure devices are randomly selected among the n devices. When the number of failed nodes is accumulated up to a threshold γ, $n - \gamma \geq k$, the repair process is triggered, and γ new devices will join the system. Therefore n-γ active storage nodes will broadcast packets to the newcomers. Note that all devices are fully connected, the newcomers can be selected randomly and uniformly from the devices which has not cached data before.

We implement our proposed schemes, including the optimal solution by solving the ILP formulation with existing solver (e.g., Lingo) for small scale scenarios, which is time consuming, and the heuristic solution using random linear coding. We compare the performance of our scheme with cooperative regenerating codes mentioned in [9], which considers cooperative repair when multiple nodes failed. In order to show the repair cost in wireless network, we use the number of transmissions as performance metric as in [9-11]. The communication cost is implied in the number of transmissions for repair, and the communication cost increases with the increase of number of transmissions for repair. For each simulation setting, we present the average performance of 200 runs.

Figure 5 illustrates the performance of our WDRC scheme. Figure 5(a) shows the impact of network size *N* on the repair cost which is measured by the number of transmissions for *c*=3, γ=*k*. It is observed that with the increasing of network size *N*, the repair cost in terms of the number of transmissions increases, as expected. The reason is that the number of failed devices is γ=*k,* and *k* is $O(\sqrt{N})$, then γ increases with increasing of *N*, and more failed devices generally requires more repair costs. Figure 5(b) shows the performance for *c*= 5, γ=*k*/2. Compared to Figure 5(a), as the number of failed devices γ decreases and *c* increases, the superiority of the proposed method decreases in Figure 5(b), since the superiority decrease of the proposed method induced by the decrease of γ is more than the superiority increase induced by increase of *c*. As Figure 5 shows, our solution reduces the number of transmissions significantly as compared to traditional cooperative regenerating codes due to the utilization of wireless broadcast nature in the proposed repair algorithm. It is also observed that the performance gap between our heuristic solution and optimal solution is small, which shows the efficiency of our proposed solution.

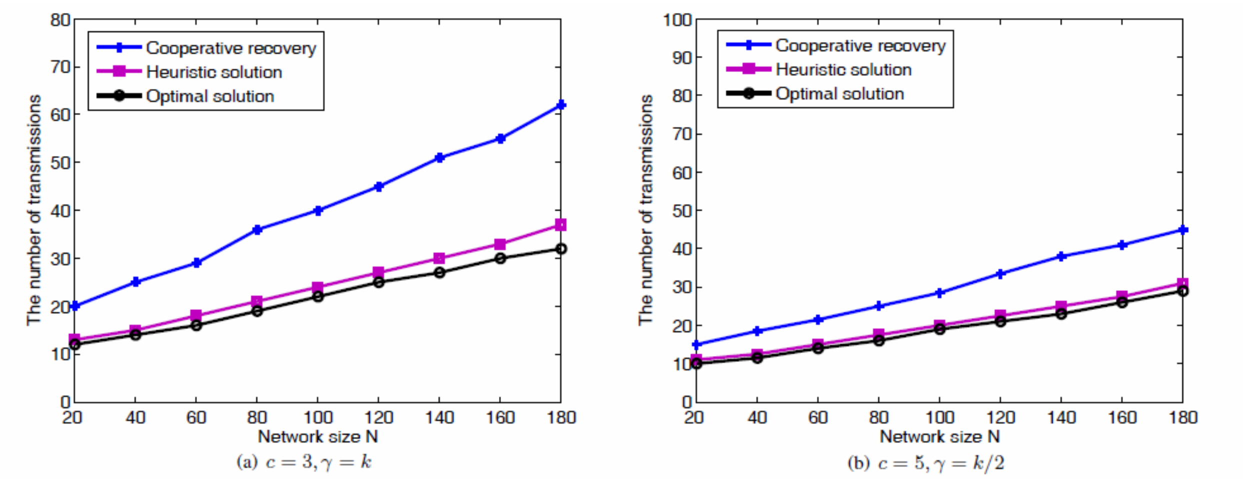

Figure 5. The performance of WDRC for different settings

6 Conclusion

In this paper, we consider the repair problem in wireless network when multiple devices that stores data fail or leave the network. By exploiting the wireless broadcast nature, we develop the constraints conditions for the repair problem using an information flow graph model, based on which we formulate the repair problem using an integer linear programming formulation. We also study the construction of repair codes and propose a decentralized repair coding method. Simulation results show that the performance of our proposed method outperforms that of traditional cooperative repair scheme, which is the basic repair method for wired distributed storage systems.

Acknowledgements

This work was supported by Fundamental Research Funds for the Central Universities (No. XDJK2019 C084), and National Natural Science Foundation of China (No. 61702426).

References

[1] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update (2013-2018)*, February, 2014.

[2] M. Huang, C. Yang, H. Li, J. Shen, Sparse Selective Encryption for HEVC 4K Video Using Spatial Error Spread, *Journal of Internet Technology*, Vol. 20, No. 5, pp. 1589-1600, September, 2019.

[3] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, G. Caire, Femtocaching: Wireless Content Delivery Through Distributed Caching Helpers, *IEEE Transactions on Information Theory*, Vol. 59, No. 12, pp. 8402-8413, December, 2013.

[4] E. Daniel, N. A. Vasanthi, ES-DAS: An Enhanced and Secure Dynamic Auditing Scheme for Data Storage in Cloud Environment, *Journal of Internet Technology*, Vol. 21, No. 1, pp. 173-182, January, 2020.

[5] N. Golrezaei, P. Mansourifard, A. F. Molisch, A. G. Dimakis, Base-station Assisted Device-to-device Communications for High-throughput Wireless Video Networks, *IEEE Transactions on Wireless Communications*, Vol. 13, No.7, pp. 3665-3676, July, 2014.

[6] M. Ji, G. Caire, A. Molisch, Fundamental Limits of Caching in Wireless D2D Networks, *IEEE Transactions on Information Theory*, Vol. 62, No. 2, pp. 849-869, February, 2016.

[7] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. J. Wainwright, K. Ramchandran, Network Coding for Distributed Storage Systems, *IEEE Transactions on Information Theory*, Vol. 56, No. 9, pp. 4539-4551, September, 2010.

[8] A. G. Dimakis, K. Ramchandran, Y. Wu, C. Suh, A Survey on Network Codes for Distributed Storage, *Proceedings of the IEEE*, Vol. 99, No. 3, pp. 476-489, March, 2011.

[9] K. W. Shum, Y. Hu, Cooperative Regenerating Codes, *IEEE Transactions on Information Theory*, Vol. 59, No. 11, pp. 7229-7258, November, 2013.

[10] J. Pedersen, A. Graell i Amat, I. Andriyanova, F. Brannstrom, Repair Scheduling in Wireless Distributed Storage with D2D Communication, *IEEE Information Theory Workshop (ITW)*, Jeju, South Korea, 2015, pp. 69-73.

[11] M. Gerami, M. Xiao, M. Skoglund, Partial Repair for Wireless Caching Networks with Broadcast Channels, *IEEE Wireless Communications Letters*, Vol. 4, No. 2, pp. 145-148, April, 2015.

[12] A. Pyattaev, O. Galinina, S. Andreev, M. Katz, Y. Koucheryavy, Understanding Practical Limitations of Network Coding for Assisted Proximate Communication, *IEEE Journal on Selected Areas in Communications*, Vol. 33, No. 2, pp. 156-170, February, 2015.

[13] P. Gupta, P. R. Kumar, The Capacity of Wireless Networks, *IEEE Transactions on Information Theory*, Vol. 46, No. 2, pp. 388-404, March, 2000.

[14] S. Jaggi, P. Sanders, P. A. Chou, M. Effros, S. Egner, K. Jain, L. Tolhuizen, Polynomial Time Algorithms for Multicast Network Code Construction, *IEEE Transactions on Information Theory*, Vol. 51, No. 6, pp. 1973-1982, June, 2005.

[15] F. H. Fitzek, T. Toth, A. Szabados, M. V. Pedersen, D. E. Lucani, M. Sipos, H. Charaf, M. Medard, Implementation and performance Evaluation of Distributed Cloud Storage Solutions Using Random Linear Network Coding, *IEEE International Conference on Communications Workshops (ICC)*, Sydney, Australia, 2014, pp. 249-254.

Biographies

Cheng Zhan received the Ph.D. degree in computer science from the University of Science and Technology of China in 2011. He is currently in the School of Computer and Information Science, Southwest University, China. His research interests include network coding, wireless network optimization, multimedia transmission and distributed storage.

Zhe Wen was born in Anhui Province, China. He is an undergraduate student in the School of Computer and Information Science, Southwest University, Chongqing, China. His research interests include network coding, mobile multimedia transmission.